**Pandas-1**

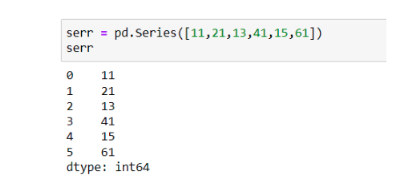
Mark as done

**Introduction to Pandas**

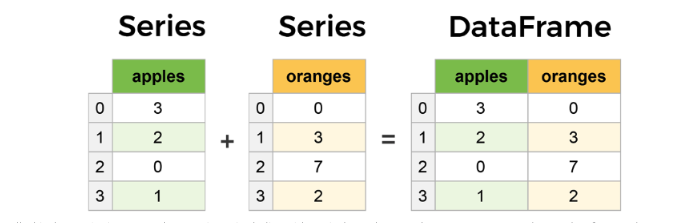
Welcome to the pandas lesson. In this lesson, you will learn what pandas is and how to install it.   
  
Pandas is a free software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. Pandas is mainly used for machine learning in form of DataFrames. Pandas allows importing data of various file formats such as csv, excel etc. And it provides various data manipulation operations such as groupby, join, merge, melt, concatenation as well as data cleaning features such as filling, replacing or imputing null values.



[What is Pandas? Why and How to Use Pandas in Python](https://www.youtube.com/watch?v=dcqPhpY7tWk)

****

Pandas series is a one-dimensional labeled array capable of holding data of any type (integer, string, float, python objects, etc.). In order to fully understand DataFrames, you need to know the basics of series. You can think of the pandas series as a column with labels in an excel sheet.

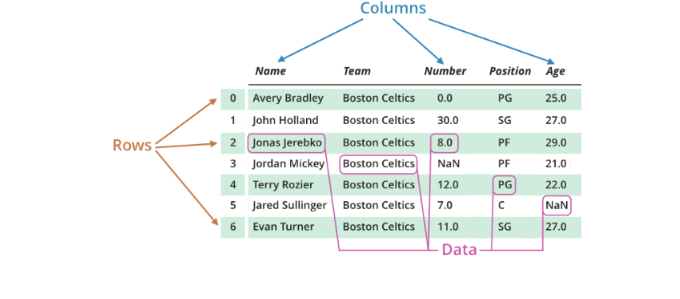


In a series, the axis labels are called indexes. Series can only contain a single list with an index, whereas the DataFrames can be made of more than one series.

[Introduction to Pandas and Dataframes | Python Pandas Tutorial #1 | Create Dataframe & Read from Web](https://www.youtube.com/watch?v=TKj0mjmSVgQ)

**Data Frame Basics-1 (Attributes)**

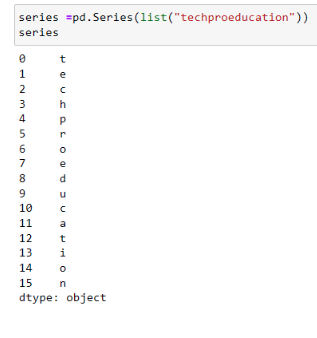
A DataFrame is a two-dimensional data structure where data is aligned in rows and columns. Three principal components; the data, rows, and columns form the Pandas DataFrame.



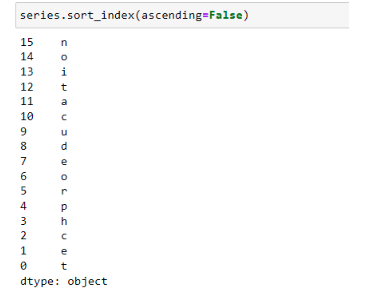
In this lesson you will learn:

* **DataFrame**(): Two-dimensional, size-mutable, potentially heterogeneous tabular data. Data structure also contains labeled axes (rows and columns). The primary pandas data structure

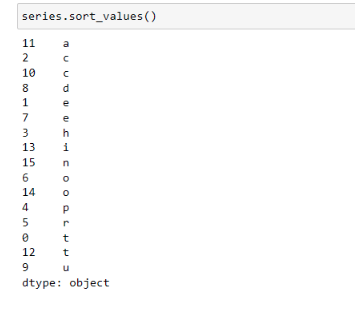
* **Series**(): One-dimensional ndarray with axis labels (including time series)



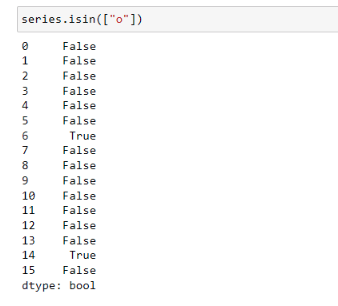
* **.sort\_index :**Sort Series by index labels

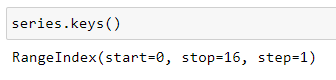


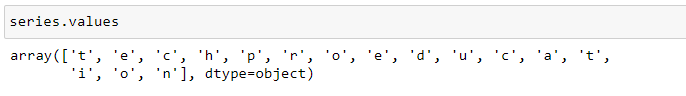
* **.sort\_values :**Sort a Series in ascending or descending order by the values



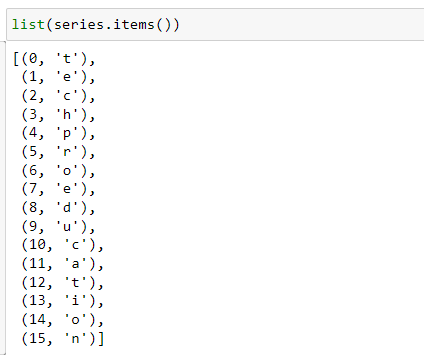
* **.isin :**Return a boolean Series showing whether each element in the Series matches an element in the passed sequence of *values* exactly



* **.keys :**return the index labels of the given series object  
  
* **.values :**Return Series as ndarray or ndarray-like depending on the dtype



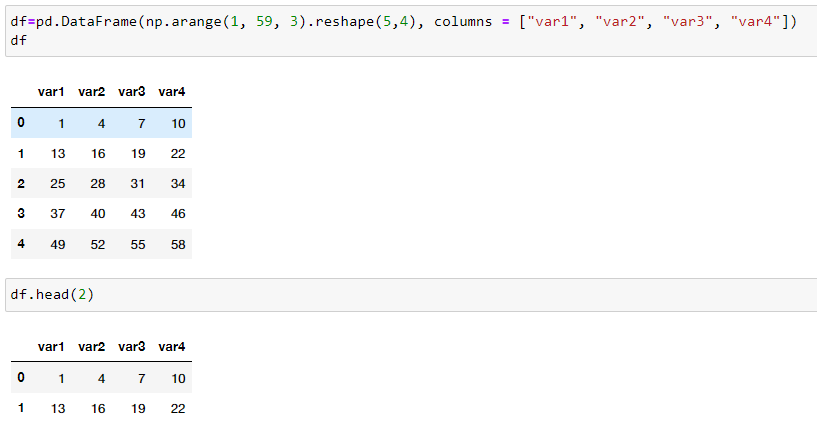
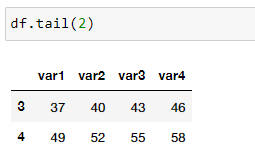
* **.items :**This method returns an iterable tuple (index, value)



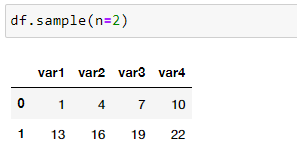
* .**read\_csv**(): Read a comma-separated values (csv) file into DataFrame



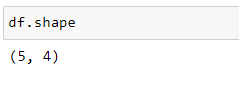
* .**head**(): This function returns the first *n* rows for the object based on position, default n=5

  
  
  
.**tail**(): This function returns last *n* rows from the object based on position, default n=5.  
  
  


* **.sample :**Return a random sample of items from an axis of object



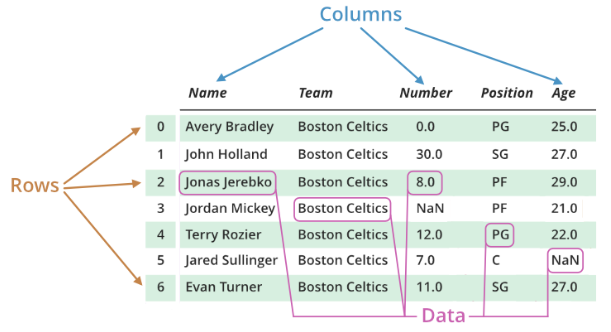
* .**shape**: Return a tuple representing the dimensionality of the DataFrame.

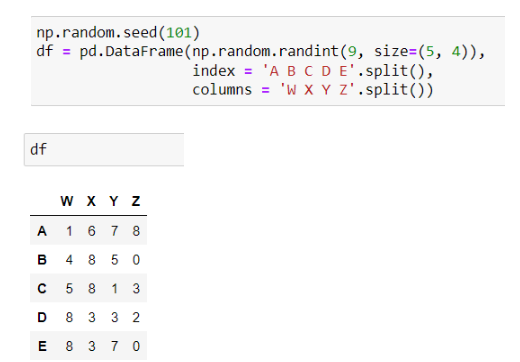


[Introduction to Pandas and Dataframes | Python Pandas Tutorial #1 | Create Dataframe & Read from Web](https://www.youtube.com/watch?v=TKj0mjmSVgQ)

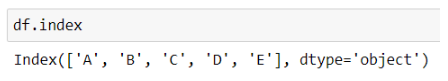
**Data Frame Basics-2 (Index & Selecting)**

A DataFrame is a two-dimensional data structure where data is aligned in rows and columns. Three principal components; the data, rows, and columns form the Pandas DataFrame.

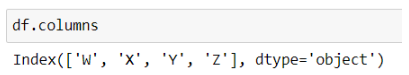




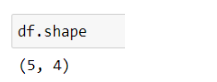
* **df.index :** The basic object storing axis labels for all pandas objects.



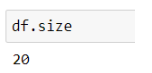
* **df.columns :** The column labels of the DataFrame.



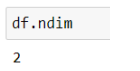
* **df.shape :** Return a tuple representing the dimensionality of the DataFrame.



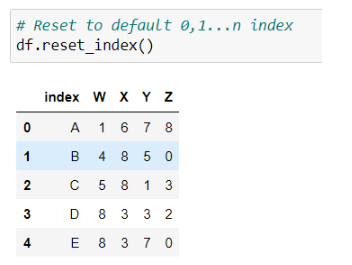
* **df.size :** Return an int representing the number of elements in this object.



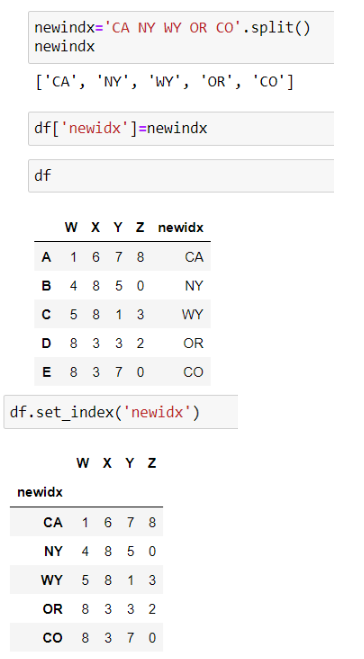
* **df.ndim :** Return an int representing the number of axes / array dimensions.



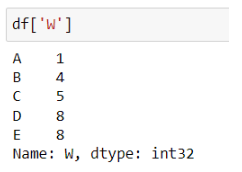
* **df.reset\_index :** Reset the index of the DataFrame, and use the default one instead. If the DataFrame has a MultiIndex, this method can remove one or more levels.



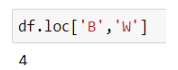
* **df.set\_index():** Set the DataFrame index using existing columns.



* **df["col"]:**You can pass a list of columns to [] to select columns in that order. If a column is not contained in the DataFrame, an exception will be raised.

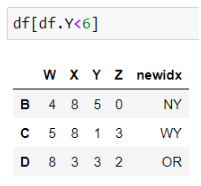


* **df.iloc[]:**Purely integer-location based indexing for selection by position.



* **Conditional Indexing:**The condition inside the selection brackets

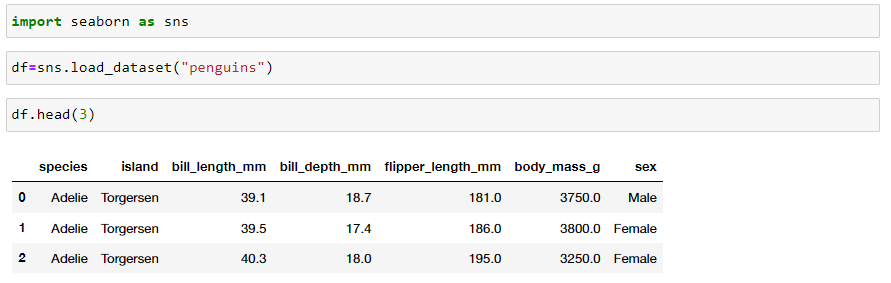
df[ df["Y"]<6 ] checks for which rows the Y column has a value smaller than 6:



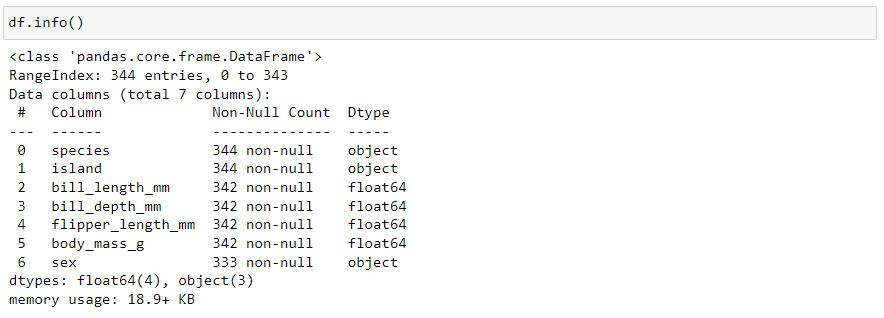
[Pandas Dataframe Index & Selecting Data | Python Pandas Tutorial #2 | iloc loc isin Pandas Function](https://www.youtube.com/watch?v=VIa1ETYnFuc)

**Data Frame Basics-3 (Properties)**

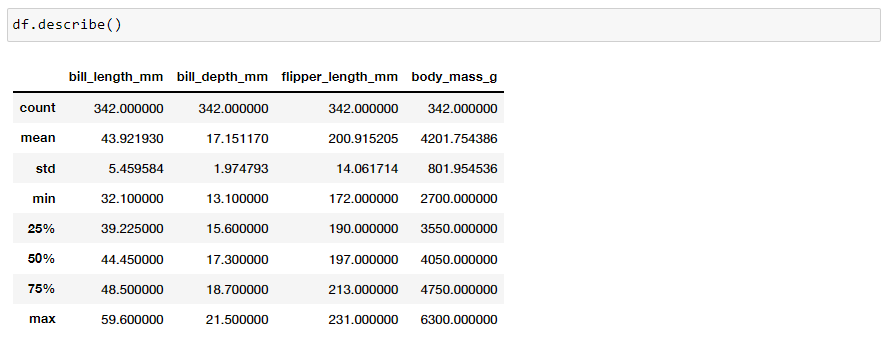
In this lesson, you will continue with DataFrame operations in Pandas.



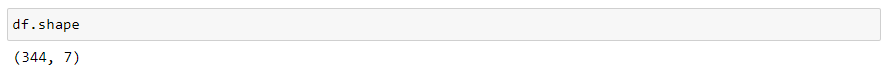
* df.info(): This method prints information about a DataFrame including the index dtype and columns, non-null values and memory usage



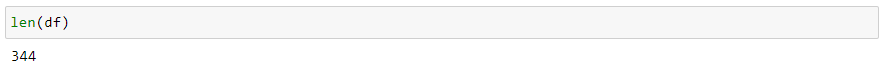
* df.describe(): Generate descriptive statistics which include those that summarize the central tendency, dispersion and shape of a dataset’s distribution, excluding NaN values



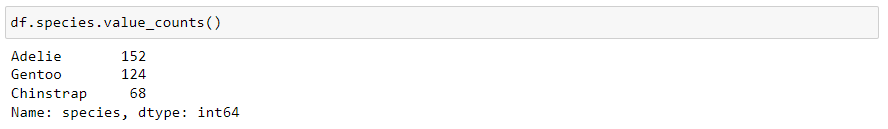
* df.shape: Return a tuple representing the dimensionality of the DataFrame.



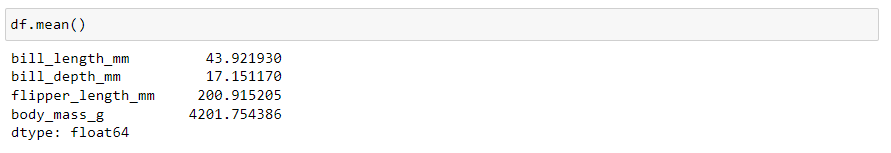
* len(df): Returns the length (number of characters) in a string. Returns the number of entries for dictionaries, lists or tuples.



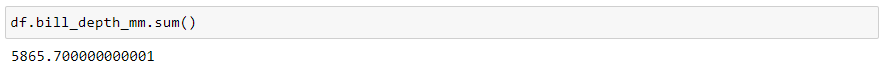
* df.col.value\_counts(): Return a Series containing counts of unique values.



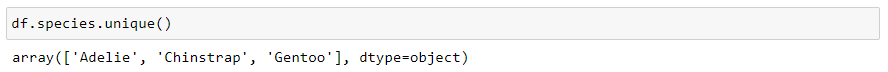
* df.mean(): Return the mean of the values over the requested axis.



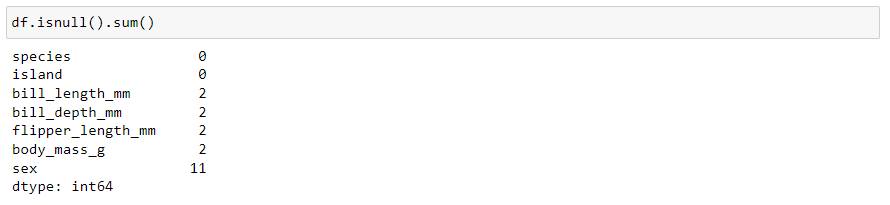
* df.col.sum(): Return the sum of the values over the requested axis.



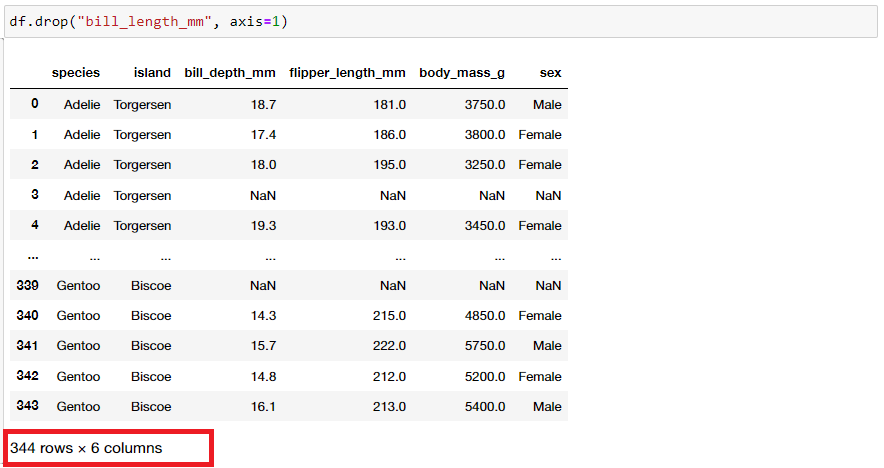
* df.col.unique(): Hash table-based unique. Uniques are returned in order of appearance. This does NOT sort

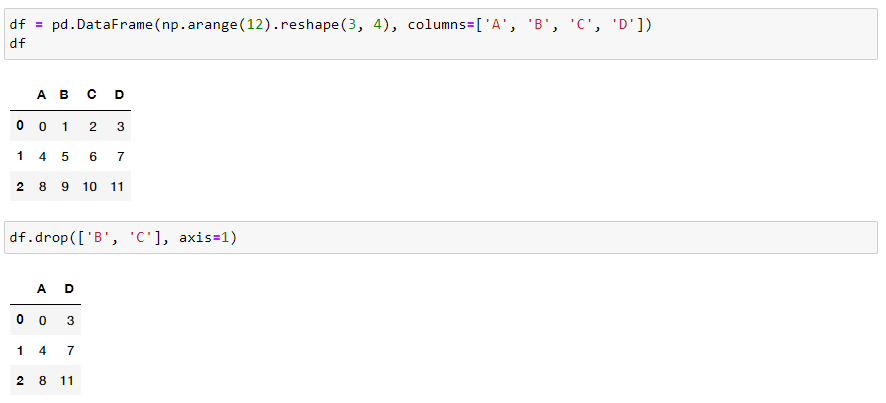


* df.isnull().sum(): Detect missing values. Return a total number of missing values in each column.



* df.drop():Remove rows or columns by specifying label names and corresponding axis, or by specifying directly index or column names





[Pandas Dataframe Basics | Python Pandas Tutorial #3 | Pandas Describe, Info, isnull, Len Functions](https://www.youtube.com/watch?v=yq9Art2Yu54)

This lesson is not ready to be taken.